

**VŠB - Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**  
**Katedra Informatiky**

**Informační systém překladatelské**  
**agentury**  
**Information system translation agency**

2010

Karel Novotný



## **Prohlášení**

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

Datum odevzdání:

Podpis:

## **Poděkování**

Děkuji vedoucímu bakalářské práce, panu doc. Ing. Lačezaru Ličevovi, Csc. za ochotnou pomoc, odborné vedení a podporu při tvorbě bakalářské práce.

## **Abstract**

### **Informační systém překladatelské agentury**

Cílem této bakalářské práce je vytvoření databázového informačního systému pro překladatelskou agenturu. Zabývá se kompletním cyklem vývoje, počínaje zadáním a návrhem, až po implementaci, testování a nasazení do provozu. Systém bude evidovat zákazníky, zaměstnance, překladatele a objednávky, které tito pomocí systému sjednali. Jejím hlavním cílem je prozkoumat rozsah a možnosti nasazení takového systému do praxe. Vzhledem k tomu, že se jedná o velmi rozsáhlou oblast, je tento systém zaměřen pouze na vytvoření objednávky a sledování jejího životního cyklu. Práce je rozdělena do 4 částí, přičemž první část se pokouší objasnit problematiku překladů, druhá část nám nastiňuje analýzu daného informačního systému, třetí část jeho implementaci a čtvrtá jeho testování.

**Klíčová slova:** informační systém, překladatelská agentura, jazyk, překladatel, zákazník, zaměstnanec, uživatel

### **Information system of a translation bureau**

This bachelor thesis aims to create an information system suitable for a translation bureau. It covers complete process of development ranging from specification and project design to implementation, testing and commission. The system will record customers, employees, interpreters and orders filed through the system.

Main purpose of the thesis is to assess practical potential of such information system. Due to comprehensive nature of the topic, it focuses on a specific part of this process: creation of order and tracking during its life cycle.

The paper is divided into 4 parts. The first one provides introduction into matters of translation, next one covers analysis of a particular information system, third part its implementation and the last one testing.

**Key words:** information system, translation bureau, language, interpreter, customer, employee

# Obsah

Úvod.....	6
Zpracování.....	7
1. Překladačské agentury a jejich činnost.....	7
1.1 Překlady.....	7
1.2 Překladačelé.....	7
1.3 Překladačské agentury.....	8
1.4 Budoucnost překladačských agentur.....	8
2. Návrh informačního systému.....	9
2.1 Rozsáhlost systému.....	9
2.2 Role.....	9
2.2.1 Zjištění rolí.....	9
2.2.2 Práva rolí.....	10
2.3 Evidovaná data.....	11
2.3.1 Výčet tabulek.....	11
2.3.2 Konkrétní rozbor tabulek.....	11
2.4 Datová analýza.....	15
2.4.1 Use Case.....	15
2.4.2 Lineární zápis typů entit.....	16
2.4.3 ER diagram.....	17
2.4.4 Datový slovník.....	18
2.4.5 DFD.....	19
2.4.6 Sekvenční diagramy.....	20
2.5 Volba prostředků.....	21
3 Implementace.....	22
3.1 Návrh webových stránek.....	22
3.2 Volba MasterPages.....	22
3.3 Zajištění autorizace.....	23
3.4 Jednotlivé stránky.....	23
3.5 Databáze .....	24
3.5.1 Vytvoření databáze.....	24
3.5.2 Mapování databáze.....	24
3.6 Logická vrstva informačního systému.....	26
3.6.1 Zabezpečení proti lidským chybám.....	26
3.6.2 Odesílání emailů.....	27
3.7 Prezenční vrstva.....	28
4 Ověření funkčnosti systému.....	29
Závěr.....	30
Přílohy.....	31

# Úvod

Úkolem mé práce bylo vytvořit informační systém překladatelské agentury. Tento systém má umožnit zákazníkům vytváření a sledování objednávek. Dále má umožnit zaměstnancům agentury řízení životního cyklu objednávky. A v neposlední řadě by měl umožnit i vyjádření k objednavce překladatelům.

V úvodu bych proto rád nastínil problematiku překladů, překladatelů a překladatelských agentur. Poté bych lehce shrnul analýzu daného informačního systému. Dále bych rád nastínil obsah implementace a testování tohoto systému.

Nejdříve bych se rád pokusil nastínit problematiku překladů. V první řadě si musíme uvědomit, co to překlad je, kdo ho využívá a jak probíhá. Překladem rozumíme přeložení textu, či mluveného slova, z jednoho jazyka do jazyka jiného, přičemž by měla zpráva a její smysl zůstat nezměněn. Abychom tohoto docílili, potřebujeme využít služeb překladatele, který je s daným jazykem dobře obeznámen, umí jej používat a v neposlední řadě zná i kulturu lidí, kteří jím hovoří.

Překlady jsou různých typů. Nejjednodušší je dělení překladů na překlady textů a překlady mluveného slova, neboli tlumočení. Každý z nich obnáší své problémy a úskalí. Mezi další možnosti dělení například patří překlady bez ověření a ověřené. Jak již přímo z názvů vyplývá, rozdíl mezi nimi je v ověření, neboli závaznosti překladu. Pokud je totiž překlad ověřený, je možno jej použít i při úředním jednání a z toho plynou nároky na něj.

Nejen překlady jsou různé, ale i překladatelé se velmi liší. Pokud provneme překladatele, který je rodilým mluvčím s překladatelem, který v dané oblasti nikdy nežil zjistíme, že první provede překlad i se zapojením znalosti dané kultury a je schopen věrněji přenést informaci, kdežto překladatel neznalý tohoto provede pouze doslovný překlad, který může být, bohužel, do jisté míry zavádějící.

Nyní bych rád objasnil pojem překladatelská agentura. Překladatelská agentura je firma, která nabízí zprostředkování překladatelských služeb zákazníkům. Aby toto mohla plnit, musí mít kontakt na jednotlivé překladatele a pokud možno mít s nimi dohodnuté podmínky, za jakých budou překlady probíhat. Jejím dalším úkolem je vyřizovat agendu ohledně překladů. Do tohoto bych zahrnul, dodávání dokumentů, schopnost pomoci vybrat pravý překlad a v neposlední řadě dát zákazníkovi jistou míru garancí ohledně překladu.

Dalším bodem je analýza daného odvětví a z toho vyplývající návrh informačního systému. V této části, která je podrobně rozepsána níže, se zabývám požadavky a jejich optimálním plněním. K tomuto používám jednak slovní vysvětlení a zároveň i diagramy, které jsou pro toto určeny. Příkladem těchto diagramů mohou být USE CASE, ER diagramy, stavové diagramy, datové slovníky a podobně.

Po této fázi přichází vlastní implementace. V této části se snažím popsat postup tvorby informačního systému na „fyzické“ úrovni. Její součástí je příprava databáze, návrh mapování, vlastní struktura webových stránek a podobně.

Poslední fází nedílně spjatou s implementací je fáze testování. V této fázi se pokusím shrnout poznatky a poskytnout doporučení pro další testování systému.

# Zpracování

## 1. Překladačské agentury a jejich činnost

### 1.1 Překlady

V dnešním světě, kdy cestování po světě je otázkou pár hodin, kdy stěhování za prací je na denním pořádku a kdy jedna firma má pobočky v různých částech světa, je jedním z velkých problémů vzájemné porozumění. Na světě existuje nepřehledné množství jazyků a nářečí, proto není v lidských silách znát je všechny a i pokud se daný jazyk naučíme, tak nikdy nedosáhneme úrovně rodilých mluvčích. Proto potřebujeme někoho, kdo zná daný jazyk a kdo je schopen nám ho „přeložit“ do jazyka, kterému rozumíme. Těmto lidem říkáme překladatelé.

Existuje celá řada překladů. Jak již bylo zmíněno výše, jedná se o překlady textů, mluveného slova, překlady ověřené a bez ověření, překlady s kontrolou překladu a podobně. A i tyto se dají dělit dále. Například podle oborů, kterých se překlad týká.

Obory překladů můžeme chápat jako odvětví. Přerovnejme si překlad z oboru informatiky a z oboru školství. Pokud přejdeme fakt, že hlavně v prvním odvětví často některé výrazy ani nelze rozumě přeložit, dojdeme ke zjištění že překladatele musí mít zcela rozdílnou slovní zásobu a znalosti.

### 1.2 Překladatelé

Tito lidé se věnují jazykům na profesionální úrovni a proto jejich služeb využívají lidé, kteří potřebují přeložit text, či mluvené slovo, co nejpřesněji. Stejně jako mezi překlady je rozdíl i mezi překladateli.

Překladatelé jak již bylo zmíněno výše mají znalost pouze omezeného množství jazyků. Dále bylo zmíněno, že i pokud znají daný jazyk, potřebují i znalost z daného oboru, ve kterém budou překlad provádět. Dalším rozdílem je znalost dotyčné kultury. Pokud bychom chtěli srovnat překlad člověka, který sice jazyk umí na maximální možné úrovni s překladem, který provede rodilý mluvčí, zjistíme, že rodilý mluvčí bude vždy překládat přesněji. Toto je způsobeno jeho znalostí místní mentality, slangu, ale i vžitých slovních spojení.

Ukázka překladu:

1995 wurden die ersten Grenzen abgebaut, einerseits zwischen den Schengen-Gründungsmitgliedern Belgien, Deutschland, Frankreich, Luxemburg und den Niederlanden sowie andererseits auch zu Spanien und Portugal.

*originální text*

V roce 1995 byly odstraněny první hranice, jednak mezi zakládajícími zeměmi Schengenského prostoru – Belgií, Německem, Francií, Lucemburskem a Nizozemskem – jednak hranice se Španělskem a Portugalskem.

*doslovný překlad*

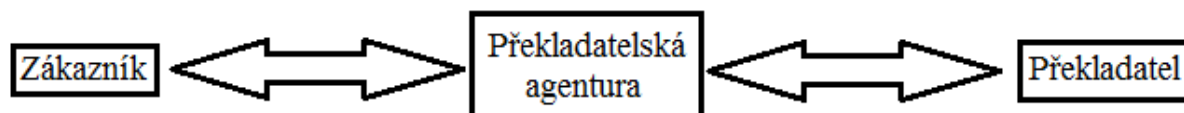
První hranice byly zrušeny v roce 1995, a to na jedné straně mezi zakládajícími zeměmi Schengenského prostoru – Belgií, Německem, Francií, Lucemburskem a Nizozemskem – současně také ve Španělsku a Portugalsku.

*volný překlad s ohledem na jeho použití*

### 1.3 Překladatelské agentury

Z předchozího vyplývá, že aby byl překladatel dobrý a jeho překlad měl co nejvyšší úroveň, je nutné aby se věnoval co nejmenšímu množství jazyků. Proto potřebujeme těchto překladatelů velké množství.

Nyní je otázkou, jak se může člověk, který má zájem o překlad, spojit s překladatelem, kterého potřebuje. K tomuto slouží překladatelské agentury. Tyto agentury jsou prostředníkem mezi zákazníkem a překladatelem. Jejich hlavním cílem je zprostředkovat překlad pro zákazníka. Aby byly schopny toto plnit, potřebují mít velké množství překladatelů. Pokud tedy shrneme předchozí tak zjistíme, že překladatelská agentura sama překlady neprovádí, ale využívá k tomu služeb překladatelů. Samozřejmě někteří z nich mohou být jejich zaměstnanci, ale jiní pracují pouze jako externisté, či samostatné osoby.



*Vztahy mezi zákazníkem, agenturou a překladatelem*

### 1.4 Budoucnost překladatelských agentur

Mnoho lidí si jistě myslí, že překlady prováděné lidmi brzy odejdou do ústraní a jejich místo převzme stroje. Vždyť již v dnešní době je k dispozici nepřeberné množství nástrojů, které nám překlad obstarají. Problémem u těchto nástrojů však navždy bude provést správný překlad v daném kontextu. Vždyť každý z nás zná slova, která mají více významů. Nebo jistě často používá víceslovná spojení, která v daném kontextu znamenají něco jiného, než jaký je význam slov v nich použitých. Proto bude vždy potřeba lidský faktor pro zajištění maximálně správného překladu. A dokud bude překlady obstarávat člověk, bude vždy potřebný mezičlánek v podobě překladatelské agentury.



## 2. Návrh informačního systému

### 2.1 Rozsáhlost systému

Při návrhu informačního systému jsem se rozhodl postupovat podle jednoho ze základních bodů standardu ITIL a tím je: „*Zákaznický orientovaný přístup*„. Tento bod nám v jednoduchosti říká, že všechny procesy se navrhují s ohledem na zákazníka. To znamená, že cokoliv co přidáme, by mělo mít pro zákazníka nějakou užitnou hodnotu.

Co tedy zákazník vyžaduje? Možnost získání informací o nás. K tomu by stačila jednoduchá webová prezentace, kterých také velmi mnoho překladatelských agentur využívá. Jaký další požadavek by mohl zákazník mít? Především možnost objednání služeb pomocí Internetu. Zde již se musíme rozhodnout, jestli chceme dělat složitou webovou prezentaci a nebo již vybudujeme informační systém. A nakonec si musíme uvědomit, že pokud si zákazník překlad objedná pomocí Internetu, tak bude jistě velmi rád, pokud bude pomocí něj moci sledovat i stav svých objednávek. A v tomto bodě je již rozhodnuto o vybudování informačního systému.

Zatím jsme zjistili, že náš informační systém by měl obsahovat webovou prezentaci, možnost vytvoření objednávky a její sledování. Nyní se dostáváme k problematice překladů.

Překladatelské agentury jsou většinou firmy, které mají jak své „interní“, překladatele, tak i překladatele, kteří nejsou jejich zaměstnanci. A zákazník jistě nebude příliš šťasten, pokud se jeho překlad opozdí kvůli špatné komunikaci s překladatelem. Proto je nutné do informačního systému zahrnout i překladatele.

Proto jsem se rozhodl vybudovat informační systém, který má na starosti prezentaci nabídky služeb, vytvoření objednávky, její sledování, přeposílání objednávek překladateli, který se k nim zároveň pomocí našeho informačního systému vyjadřuje. A samozřejmě o všech těchto bodech informuje uživatele systému, kterých se to týká.

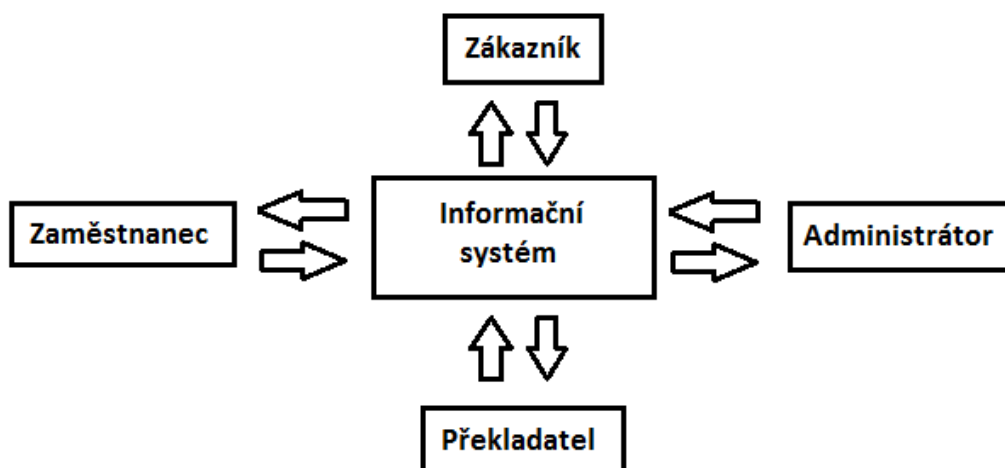
### 2.2 Role

#### 2.2.1 Zjištění rolí

Dalším krokem je určit jednotlivé uživatele systému. Uživatelem bude jistě zákazník, pro kterého tento systém budujeme. Dalším uživatelem bude překladatel, který defakto zajišťuje naši nabízenou službu. Jistě potřebujeme i mezičlánek mezi těmito typy uživatelů. A tím bude zaměstnanec agentury. Tímto výčtem jsme pokryli hlavní aktéry procesu objednávání a plnění objednávky, avšak nesmíme zapomenout na roli správce. Vždy musí existovat někdo, kdo vytváří nové účty a nebo je naopak maže. Tímto nám vznikla čtvrtá role a to role administrátorská.

Pokud si tedy shrneme tento bod, tak z něj vyplývá, že náš systém bude mít 4 typy účtů, kterými jsou: *zákazník, zaměstnanec, překladatel a administrátor*.

Okolí systému tedy bude vypadat takto:



### 2.2.2 Práva rolí

Nyní je nutné zjistit, jaké práva musí mít jednotlivé role:

- Zákazník, pro kterého je tento informační systém určen primárně, musí mít možnost vytvoření objednávky a její sledování. Co dalšího by zákazník mohl uvítat? Především být neustále informován o změnách v jeho objednávce. Proto jistě do našeho systému implementujeme posílání zpráv o změně zakázky.
- Zaměstnanec musí mít přístup k objednávkám, za které zodpovídá. Dále musí mít možnost měnit jejich stav. I u této role bude velmi výtanečné upozornění na změnu. Jako nejjednodušší se jeví opětovné použití upozornění pomocí emailů.
- Překladatel našeho systému využívá k získání informace o nabídce objednávky, proto musí mít možnost si tyto objednávky prohlédnout. Dále se musí k daným objednávkám vyjádřit a proto mu musíme umožnit změnit stav jeho objednávek. A samozřejmě, jako veškeré předešlé role, ho musíme na změnu upozornit.
- Administrátor musí mít možnost vytvářet nové role a mazat nepotřebné. Dále by měl mít možnost smazat zakázku, která se již dále nemůže změnit. A nakonec by měl mít možnost vkládat informace potřebné pro jednotlivé role.

## 2.3 Evidovaná data

### 2.3.1 Výčet tabulek

Tím se dostáváme k dalšímu bodu. A tím je co vš se bude v našem informačním systému evidovat. Já jsem se rozhodl evidovat překlady, které přiřazuji jednotlivým překladatelům. Tím docílím toho, že zákazník si vybere překlad a k němu je předem dán překladatel. Sice to vede k problému v případě, že více překladatelů nabízí obdobný překlad, ale tento problém řeším vytvořením více stejných překladů. Překlad jako takový, proto obsahuje jméno, do kterého zapisujeme jméno zdrojového a cílového jazyka, pak obor, kterého se tento překlad týká, dále pak orientační cenu a nakonec poznámku. Odlišení tedy probíhá volbou oboru. Pokud by i zde docházelo ke střetu dvou překladatelů, tak je k dispozici poznámka. Tímto zamezíme tomu, aby více překladatelů mělo jeden typ překladu a zároveň přiblížíme systém realitě, kdy zákazníka nezajímá překladatel ale překlad který mu nabízíme.

Dále překladatelé používají různé nástroje. Proto tyto musíme také evidovat. Vzhledem k jejich omezenému počtu se jeví jako ideální způsob jejich uložení do číselníku a sami překladatelé si rozhodnou, který z nástrojů nabídnou k překladu.

Shrnutím předešlého zjistíme, že zatím potřebujeme tabulku pro zákazníky, zaměstnance, překladatele, typy překladů, nástroje a samozřejmě pro objednávky. Objednávky jsem jsem rozdělil na objednávku zákazníka zaměstnanci a objednávku zaměstnance překladateli. Dále vzhledem k nutnosti rozlišní oprávnění uživatelů budeme potřebovat tabulku i pro přihlašování.

### 2.3.2 Konkrétní rozbor tabulek

Nyní musíme rozhodnout, co ve které tabulce budeme evidovat. V této části se budeme snažit zjistit, které údaje bude vhodné zanést do našeho informačního systému a které naopak nikoliv.

a) Uživatelé systému:

Zákazník:

- id zákazníka – jednoznačná identifikace zákazníka
- jméno a příjmení
- bydliště
  - obec
  - ulice
  - číslo popisné
  - PSČ
- kontaktní údaje
  - telefon
  - email
- poznámka

Zaměstnanec:

- id zaměstnance – jednoznačná identifikace zaměstnance
- jméno a příjmení
- bydliště
  - obec
  - ulice
  - číslo popisné
  - PSČ
- kontaktní údaje
  - telefon
  - email
- poznámka

Překladatel:

- id překladatele – jednoznačná identifikace překladatele
- jméno a příjmení
- bydliště
  - obec
  - ulice
  - číslo popisné
  - PSČ
- kontaktní údaje
  - telefon
  - email
- jazyk překladu
- nástroje, které k překladu využívá
- typ – jestli je interní nebo externí
- hodnocení překladatele
- poznámka

b) Ostatní entity:

Jazyk:

- id překladu – jednoznačná identifikace
- jméno – název zdrojového a cílového jazyka překladu
- obor – obor pro, který je daný překlad určen
- cena
  - normostránka
  - u ověření cena za kus
- poznámka

Nástroj:

- id nástroje – jednoznačná identifikace
- jméno nástroje
- cena – jedná se o koeficient, kterým se násobí cena překladu
- poznámka

Zakázka zákazník – zaměstnanec:

- id objednávky – jednoznačná identifikace objednávky
- id zákazníka – jednoznačná identifikace zákazníka
- id zaměstnance – jednoznačná identifikace zaměstnance
- id jazyka ( překladu ) – jednoznačná identifikace překladu
- způsob dodání – určení jakou formou budou dokumenty dodány
- datum vyhotovení – datum, do kterého musí být překlad vyhotoven
- datum potvrzení – datum, do kterého zákazník musí znát stanovisko překladatele
- počet kusů – počet dokumentů k přeložení
- počet kopií – počet kopií překladu
- nástroj – nástroj použitý pro překlad
- odhadovaná cena – předběžná cena překladu, která se však může lišit od skutečné
- skutečná cena – skutečná cena překladu, kterou zákazník musí uhradit
- stav – jedná se o stav zakázky, tzn ve kterém stádiu se právě nachází
- poznámka

Zakázka zaměstnanec – překladatel:

- id objednávky – jednoznačná identifikace objednávky
- id překladatele – jednoznačná identifikace překladatele
- id zaměstnance – jednoznačná identifikace zaměstnance
- id jazyka ( překladu ) – jednoznačná identifikace překladu
- způsob dodání – určení jakou formou budou dokumenty dodány
- datum vyhotovení – datum, do kterého musí být překlad vyhotoven
- datum potvrzení – datum, do kterého zákazník musí znát stanovisko překladatele
- počet kusů – počet dokumentů k přeložení
- počet kopií – počet kopií překladu
- nástroje – nástroj použitý pro překlad
- odhadovaná cena – předběžná cena překladu, která se však může lišit od skutečné
- skutečná cena – skutečná cena překladu, kterou zákazník musí uhradit
- stav – jedná se o stav zakázky, tzn ve kterém stádiu se právě nachází
- poznámka

Login:

- id loginu – jednoznačná identifikace daného oprávnění pro přístup
- přihlašovací jméno – jednoznačné přihlašovací jméno do systému
- heslo – heslo, které ve spojení s přihlašovacím jménem ověřuje totožnost uživatele
- id osoby ( id dané osoby ) - id osoby, které přihlašovací údaje náleží
- role ( zákazník, zaměstnanec, překladatel, administrátor ) - role v níž uživatel vystupuje

Nyní máme za sebou analýzu projektu z lidského hlediska. V ní jsme rozhodli o rozsáhlosti systému, zjistili jednotlivé uživatele systému, přiřadili jim role v něm a určili, co má všeobecně systém uchovávat. Tato fáze je velmi důležitá a provádí se většinou přímo se zadavatelem práce. Během této činnosti se snažíme zjistit přesnou představu zadavatele a pomoci mu domyslet jednotlivé důsledky a navrhnout případně další vylepšení.

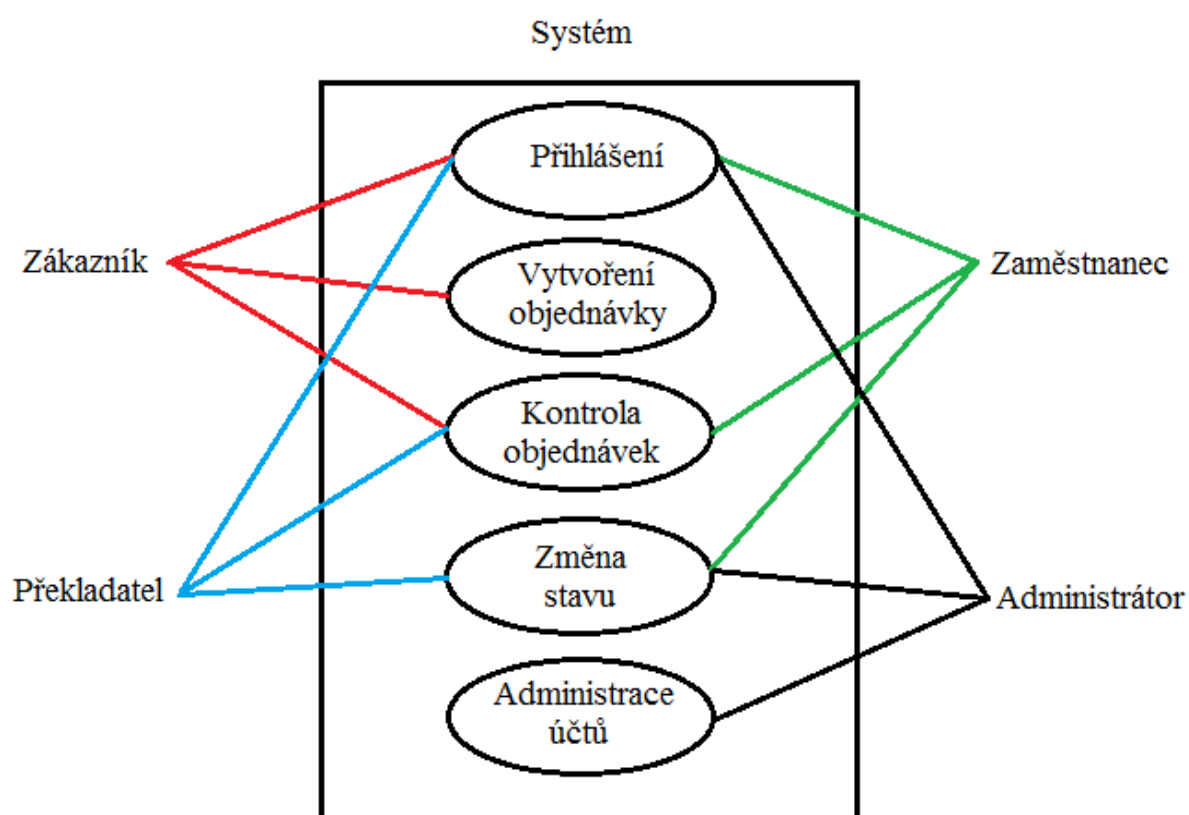
Vzhledem k tomu, že tato je zatím nedostatečná pro tvorbu systému, jsme nuceni přistoupit k analýze systému z pohledu programátora. Tato analýza se skládá z grafů a diagramů, které jednoznačně identifikují procesy, entity a jejich vlastnosti, dále řeší interakce mezi těmito entitami a jejich vývoj v čase. Její součástí jsou USE CASE diagramy, lineární zápisy typů entit, ER diagramy, datové slovníky, DFD diagramy, sekvenční diagramy a spousty dalších.

## 2.4 Datová analýza

Jejím cílem je co nejdříve zachytit realitu. K tomu nám slouží různé diagramy a tabulky.

### 2.4.1 Use Case

- jedná se o dynamické zobrazení struktury systému z pohledu uživatele
- určuje chování systému aniž by odhaloval vnitřní strukturu



## 2.4.2 Lineární zápis typů entit

- obsahuje specifikace entitních typů a vztahů mezi těmito entitními typy

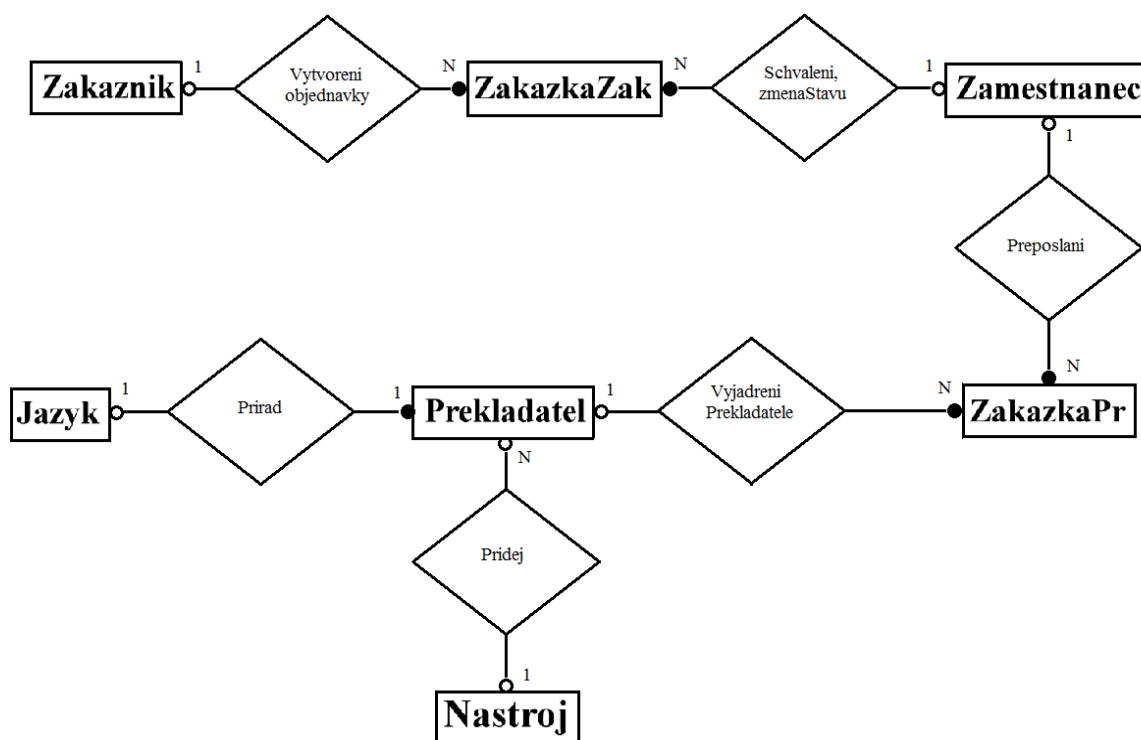
Zakaznik	( <b>id</b> , jmeno, prijmeni, obec, ulice, cp, psc, telefon, email, poznamka)
Zamestnanec	( <b>id</b> , jmeno, prijmeni, obec, ulice, cp, psc, telefon, email, poznamka)
Prekladatel	( <b>id</b> , jmeno, prijmeni, obec, ulice, cp, psc, telefon, email, <u>idJaz</u> , nastroje, typPrekladatele, hodnoceni, poznamka)
Jazyk	( <b>id</b> , jmeno, cenaMezery, cenaBez, obor, poznamka)
Nastroj	( <b>id</b> , jmeno, cena, poznamka)
ZakazkaZak	( <b>id</b> , <u>idZak</u> , <u>idZam</u> , <u>idJaz</u> , zpusobDodani, datum, datumPotvrzeni, pocetKusu, pocetKopii, odhadovanaCena, cena, nastroje, stav, poznamka)
ZakazkaPr	( <b>id</b> , <u>idPr</u> , <u>idZam</u> , <u>idJaz</u> , zpusobDodani, datum, datumPotvrzeni, pocetKusu, pocetKopii, odhadovanaCena, cena, nastroje, stav, poznamka)
Login	( <b>id</b> , login, heslo, <u>idOsoby</u> , role)

- primární klíč je označen tučně, cizí klíč je podtržen



### 2.4.3 ER diagram

- zachycuje graficky vztahy mezi entitami, jejich vazby a typy



- prázdné kolečko znamená, že tato entita nemá povinnost být v daném vztahu
- plné kolečko znamená, že tato entita musí být v daném vztahu
- vazby:
  - 1:1 – Jedna entita musí mít právě jednu entitu druhého typu a naopak.
  - 1:N – K jedné entitě může náležet více entit druhého typu, avšak opačná entita může mít pouze jednu.
  - M:N – K jedné entitě může náležet více entit druhého typu a naopak.

## 2.4.4 Datový slovník

- je to seznam všech datových objektů, které jsou reprezentovány v databázi
- obsahuje jména, datové typy, jejich omezení a podobně
- příklady pro tabulky Zakaznik, Zamestnanec, Prekladatel

Zákazník

Název	Datový typ	Délka	NULL	Index	Key	IO	Poznámka
id	int	100	N	Y	Y		PK
jmeno	varchar	100	N	N	N		
prijmeni	varchar	100	N	Y	N		
obec	varchar	100	N	N	N		
ulice	varchar	100	Y	N	N		
cp	varchar	100	N	N	N		
psc	varchar	100	N	N	N		
telefon	varchar	100	Y	N	N		
email	varchar	100	N	N	N		
poznamka	varchar	1000	Y	N	N		

Zaměstnanec

Název	Datový typ	Délka	NULL	Index	Key	IO	Poznámka
id	int	100	N	Y	Y		PK
jmeno	varchar	100	N	N	N		
prijmeni	varchar	100	N	Y	N		
obec	varchar	100	N	N	N		
ulice	varchar	100	Y	N	N		
cp	varchar	100	N	N	N		
psc	varchar	100	N	N	N		
telefon	varchar	100	Y	N	N		
email	varchar	100	N	N	N		
poznamka	varchar	1000	Y	N	N		

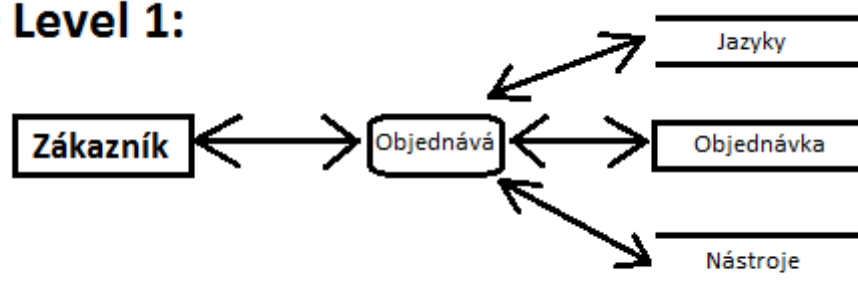
Překladatel

Název	Datový typ	Délka	NULL	Index	Key	IO	Poznámka
id	int	100	N	Y	Y		PK
jmeno	varchar	100	N	N	N		
prijmeni	varchar	100	N	Y	N		
obec	varchar	100	N	N	N		
ulice	varchar	100	Y	N	N		
cp	varchar	100	N	N	N		
psc	varchar	100	N	N	N		
idJaz	int	30	N	Y	N		FK
typPrekladatele	varchar	100	N	N	N		
telefon	varchar	100	Y	N	N		
email	varchar	100	N	N	N		
poznamka	varchar	1000	Y	N	N		
hodnoceni	varchar	100	Y	N	N		
nastroje	varchar	1000	N	N	N		cizí klíče

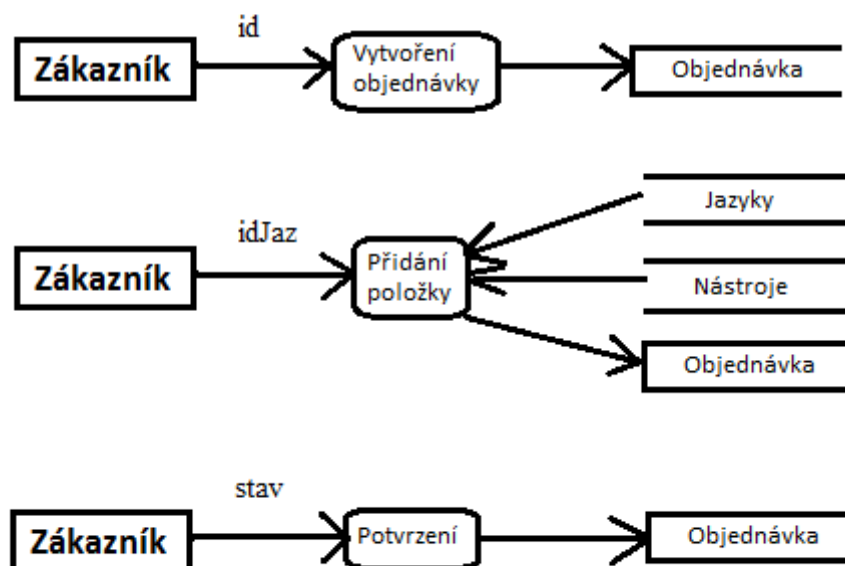
## 2.4.5 DFD

- jedná se o podrobnější ekvivalent USE CASE diagramů
- vytváří se ve více úrovních, přičemž by mělo platit, že čím vyšší úroveň, tím podrobnější diagram
- příklad DF diagramu pro vytvoření objednávky

### DFD Level 1:



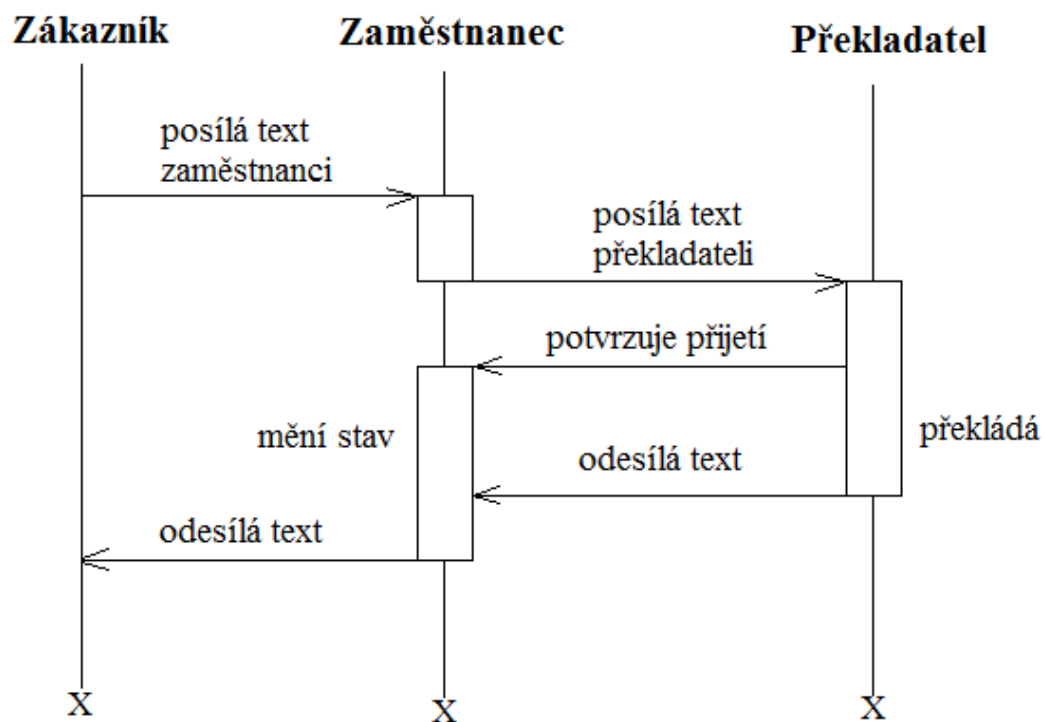
### DFD Level 2:



## 2.4.6 Sekvenční diagramy

- popisují interakci mezi objekty v závislosti na čase

Příklad:



## 2.5 Volba prostředků

Co musíme zvolit:

- programovací jazyk
  - hlavní možnosti:
    - a) Java - komplikovaný vývoj, který by v důsledku mohl systém prodražit
      - velké množství chyb ve vývojovém prostředí
    - b) PHP - v dnešní době, dle mého názoru, již lehce zastaralý jazyk
    - c) .NET - vzhledem k jeho rozsáhlé podpoře, výbornému vývojovému prostředí jakým beze sporu je Visual studio, k vynikající dokumentaci volně přístupné na Internetu jsem se rozhodl právě pro něj
- vývojové prostředí
  - ➔ jelikož jsem se rozhodl vyvíjet informační systém na platformě .NET, rozhodl jsem se použít Visual studio 2008
- typ databáze
  - ➔ mé rozhodnutí ovlivnil hlavně programovací jazyk, ve kterém chci informační systém psát a tak jsem se rozhodl pro MS SQL

## 3 Implementace

### 3.1 Návrh webových stránek

Nejdříve si musíme rozmyslet jaké webové stránky v informačním systému budou a rozhodnout, které role potřebují vidět jakou stránku.

Jakýkoliv uživatel:

- všeobecné stránky, které slouží k informování
- možnost se přihlásit do systému

Zákazník:

- vytvořit novou objednávku
- sledovat stávající objednávky

Zaměstnanec:

- sledovat objednávky, které se ho týkají a vyjadřovat se k nim

Překladač:

- sledovat objednávky, které se ho týkají a vyjadřovat se k nim

Administrátor:

- vytvářet a mazat účty

### 3.2 Volba MasterPages

Pokud máme toto rozvržení, tak můžeme navrhnout MasterPage pro každou roli. MasterPage je šablona pro naši webovou stránku. Je to vynikající prostředek pro sjednocení vzhledu a funkčnosti.

MasterPages:

- veřejne.master
  - zobrazuje menu pro nepřihlášeného uživatele
  - zobrazuje přihlašovací tabulku
- admin.master
  - zobrazuje menu pro přihlášeného administrátora
- zakaznik.master
  - zobrazuje menu pro přihlášeného zákazníka

- `zamestnanec.master`
  - zobrazuje menu pro přihlášeného zaměstnance
- `prekladatel.master`
  - zobrazuje menu pro přihlášeného překladatele

### 3.3 Zajištění autorizace

Nyní musíme zajistit, aby nikdo nemohl přistupovat tam, kam nemá oprávnění.

K tomuto máme vytvořenou tabulku Login. Ta uchovává informace o přihlašovacím jméně a hesle každého uživatele a k tomuto je přiřazeno jeho `idOsoby` a role, pod kterou do systému vstupuje. Při přihlášení zašleme dotaz na vrácení `idOsoby` a role, které náleží k danému loginu a heslu. Toto uložíme do Session.

Nyní při každém vstupu na stránku, stačí pouze prověřit jestli na ni má dotyčná role přístup. Pro jednoduchost toto implementujeme vždy v `MasterPage`.

### 3.4 Jednotlivé stránky

#### Stránky veřejně přístupné

Na těchto stránkách probíhá prezetace firmy a jejich služeb. Přihlašování proto není potřebné. Vzhledem k tomu, že tyto stránky se mohou velmi často měnit, jsem se rozhodl umožnit firmě jejich úpravu pomocí vložení XML souboru. Do něj stačí psát text či html text, který se poté objeví na dané webové stránce.

Jedinou výjimku v tomto jsem udělal na stránce služby, kde jsem použil výpis nabízených jazyků z databáze.

#### Stránky pro zákazníka

Zde jsou pouze 2 stránky:

První pro výpis objednávek daného zákazníka, která slouží k informování o průběhu.

Druhá pro vytvoření objednávky.

#### Stránky pro zaměstnance

Zde je pouze stránka, která informuje zaměstnance o objednávkách za které zodpovídá a umožňuje mu měnit jejich stav.

## Stránky pro překladače

Také pouze jedna stránka, na které je výpis objednávek a možnost vyjádřit se k nim.

## Stránky pro administrátora

Zde jsou stránky, které umožňují vytváření a mazání účtů, nástrojů, jazyků a zakázek, které již skončily svůj životní cyklus.

## 3.5 Databáze

### 3.5.1 Vytvoření databáze

Databáze je vytvořena pomocí designeru Visual studia a je uložena přímo v naší solution. Vzhledem k ochraně databáze, která je přítomna v logické vrstvě není potřeba využívat transakce či podobné.

### 3.5.2 Mapování databáze

Mapování databáze by mělo sloužit k co nejefektivnějšímu využití a zároveň pro odstínění logické vrstvy od datové. Toto lze provést buď ručně, přičemž máme velkou kontrolu nad tímto a nebo použít některý z nástrojů, který toto udělá za nás. Vzhledem k tomu, že naše přístupy do databáze jsou ve většině případů odlišné a prakticky využitelné mapování je možné jen v několika málo možnostech, rozhodl jsem se pro mapování ruční.

Mapovací třídy jsou rozděleny na 3 typy:

- a) třídy, které mapují tabulky na objekty
  - tyto třídy obsahují proměnné, které jsou obsaženy v dané tabulce, přičemž jim přiřazujeme pokud možno co nejbližší datové typy

```
public class Prekladatel
{
    int id;
    public int Id
    {
        get { return id; }
        set { id = value; }
    }
    string jmeno;
    public string Jmeno...
```



- b) třídy, které obsahují metody pro práci s těmito tabulkami
- tyto třídy obsahují naši implementaci metod pro práci s databází
  - jedná se o metody SELECT, INSERT, UPDATE a DELETE
  - u každé implementace se snažíme co nejvíce specifikovat daný dotaz, aby zatížení databáze bylo co nejmenší

```
public class PrekladatelTable : Tabulka
{
    public static String jmeno_tabulky = "Prekladatel";
    public String SQL_SELECT = "SELECT * FROM " + jmeno_tabulky + ";";
    public String SQL_SELECT_ID = "SELECT * FROM " + jmeno_tabulky + " WHERE id=@id;";
    public String SQL_SELECT_IDJ = "SELECT * FROM " + jmeno_tabulky + " WHERE idJaz=@idJaz;";
    public String SQL_INSERT = "INSERT INTO " + jmeno_tabulky + " (id, jmeno, prijmeni, obec, ulice, cp,
        + " VALUES (@id, @jmeno, @prijmeni, @obec, @ulice, @cp, @psc, @telefon, @email, @poznámka, @idJaz, @t
    public String SQL_DELETE = "DELETE FROM " + jmeno_tabulky + " WHERE id=@id;";

    public PrekladatelTable()
    {

    }

    private void pripravPrikaz(SqlCommand prikaz, Prekladatel pr) {...}
    private Collection<Prekladatel> Read(SqlDataReader reader) {...}
```

- c) a třída, která se stará o připojení k databázi
- tato třída spravuje vlastní připojení k databázi
  - zároveň určuje pomocí jakých prostředků se mají dotazy vykonávat

```
public Database()
{
    Tabulka.SetDatabase(this);
    pripojeni = new SqlConnection();
}

public bool utvorPripojeni(String connectionString)
{
    mConnectNumber++;

    if (pripojeni.State != System.Data.ConnectionState.Open)
    {
        pripojeni.ConnectionString = connectionString;
        pripojeni.Open();
    }
    return true;
}
```

## 3.6 Logická vrstva informačního systému

Tato vrstva má na starosti vlastní funkčnost systému. Pracuje jak s logickou vrstvou, tak i s prezenční. Jsou zde také implementovány mechanismy pro zabránění pádu systému.

### 3.6.1 Zabezpečení proti lidským chybám

Náš systém se snaží být co nejméně náchylný k lidským chybám. Proto se zde staráme o to, aby uživatel měl možnost udělat pouze to, co náš systém negativně neovlivní.

Tohoto dosahujeme omezením možností jeho interakce pouze na kroky, které jsou v daném okamžiku korektní. Toto sice lehce snižuje rychlost informačního systému, který musí častěji přistupovat do databáze, avšak v rámci bezpečnosti a intuitivnosti systému je toto velmi užitečné.

Ukázka dynamické změny nabídky pro změnu v objednávce:

```
protected void vyberIdPrijateobjList_SelectedIndexChanged(object sender, EventArgs e)
{
    vyberStavPrijateObjList.Items.Clear();

    try
    {
        foreach (ZakazkaZak z in kolekcePotvrzenychZakazek)
        {
            if (z.Id == Convert.ToInt32(vyberIdPrijateobjList.SelectedValue))
            {
                switch (z.Stav)
                {
                    case "Prijato prekladatelem":
                        vyberStavPrijateObjList.Items.Add("Cekani na dokument od zakaznika");
                        break;
                    case "Cekani na dokument od zakaznika":
                        vyberStavPrijateObjList.Items.Add("Predani textu prekladateli");
                        break;
                    case "Predani textu prekladateli":
```

Na obrázku vidíme jak v závislosti na aktuálním stavu objednávky vytváříme zaměstnanci nabídku pro práci s ní. Tímto zabraňujeme chybám z nepozornosti, kdy by například mohlo dojít k přeskočení některé fáze nebo naopak k vrácení zpět.

### 3.6.2 Odesílání emailů

Dále by o každé změně objednávky měli být informováni všichni zainteresovaní. Proto je jim při potvrzení změny odeslán email s upozorněním. Email navíc posíláme při vytvoření účtu zákazníka, aby tento získal své přihlašovací údaje. Pro získání informací o smtp serveru používáme xml dokument Nastaveni. Proto při jejich případné změně stačí přepsat změněné údaje v tomto souboru a problém je vyřešen. Element default obsahuje emailovou adresu, na kterou jsou odesílány požadavky na registraci.

Ukázka Nastaveni.xml:

```
<?xml version="1.0" encoding="utf-8" ?>
<nastaveni>
  <email>
    <smtp>smtp.vsb.cz</smtp>
    <host>25</host>
    <default>nov896@vsb.cz</default>
  </email>
</nastaveni>
```

Třída pro odesílání emailů:

```
public Email(string HOST, int PORT)
{
    smtp = new SmtpClient(HOST, PORT);
    this.hlavicka = "nebyla vyplnena";
    this.zprava = "nebyla vyplnena";
}

public void Odeslat(string ODESILATEL, string PRIJEMCE)
{
    try
    {
        MailMessage m = new MailMessage(ODESILATEL, PRIJEMCE, hlavicka, zprava);
        this.smtp.Send(m);
    } catch
    {}
}

public void NastavZpravu(string HLAVICKA, string ZPRAVA)
{
    this.hlavicka = HLAVICKA;
    this.zprava = ZPRAVA;
}
```

### 3.7 Prezenční vrstva

Tato vrstva tvoří grafické rozhraní pro uživatele. Je reprezentována jednotlivými webovými stránkami, ve kterých jsou obsaženy výpisy a ovládací prvky. Pro výpis dynamických dat využívám komponenty GridView. Tato komponenta vytvoří tabulku, kterou následně naplníme z databáze a celek zobrazíme uživateli.

Příklad GridView:

ID	Jméno zákazníka	Jméno jazyka	Datum potvrzení	Datum odevzdání	Dodání	Odhadovaná cena v Kč	Cena v Kč	Počet dokumentů	Počet kopií	Stav	Poznámka
1	Karel Novotný	AJ - NJ	1.5.2010 0:00:00	2.5.2010 0:00:00	Postou	400090	0	1000	3	Zamítuto	
2	Karel Novotný	AJ - NJ	3.5.2010 0:00:00	3.5.2010 0:00:00	Osobne	800180	0	1000	3	Zamítuto	
3	Karel Novotný	AJ - NJ	30.5.2010 0:00:00	30.5.2010 0:00:00	Postou	400090	0	1000	3	Zamítuto	
4	Karel Novotný	AJ - NJ	29.5.2010 0:00:00	30.5.2010 0:00:00	Osobne	400090	0	1000	3	Zamítuto	
5	Karel Novotný	AJ - NJ	28.5.2010 0:00:00	30.5.2010 0:00:00	Elektronicky	400090	40300	1000	3	Zakazka uzavrena	

Dále využívám standartních komponent jako je Text box, Label, Button a podobně. Jejich funkce plynou přímo z jejich názvů.

Příklad využití ostatních komponent:

Kontrola objednávky:	ID objednávky:	<input type="text"/>	Vyjádření:	<input type="text" value="Zamitnuto"/>	<input type="button" value="Potvrd"/>
Postup plnění objednávky:	ID objednávky:	<input type="text"/>	stav:	<input type="text"/>	<input type="button" value="Potvrd"/>

Celková stránka je složena ze 3 částí. První část je navigace, kterou obstarává daná šablona. Druhou částí je možnost interakce se systémem. A třetí částí je výpis z databáze.

Příklad celkového vzhledu systému:

<input type="button" value="Vaše objednávky"/>	<input type="button" value="Ohodnot překladatele"/>	<input type="button" value="Odhlášení"/>									
Kontrola objednávky:	ID objednávky:	<input type="text"/>	Vyjádření:	<input type="text" value="Zamitnuto"/>	<input type="button" value="Potvrd"/>						
Postup plnění objednávky:	ID objednávky:	<input type="text"/>	stav:	<input type="text"/>	<input type="button" value="Potvrd"/>						
ID	Jméno zákazníka	Jméno jazyka	Datum potvrzení	Datum odevzdání	Dodání	Odhadovaná cena v Kč	Cena v Kč	Počet dokumentů	Počet kopií	Stav	Poznámka
1	Karel Novotný	AJ - NJ	1.5.2010 0:00:00	2.5.2010 0:00:00	Postou	400090	0	1000	3	Zamítuto	
2	Karel Novotný	AJ - NJ	3.5.2010 0:00:00	3.5.2010 0:00:00	Osobne	800180	0	1000	3	Zamítuto	
3	Karel Novotný	AJ - NJ	30.5.2010 0:00:00	30.5.2010 0:00:00	Postou	400090	0	1000	3	Zamítuto	
4	Karel Novotný	AJ - NJ	29.5.2010 0:00:00	30.5.2010 0:00:00	Osobne	400090	0	1000	3	Zamítuto	
5	Karel Novotný	AJ - NJ	28.5.2010 0:00:00	30.5.2010 0:00:00	Elektronicky	400090	40300	1000	3	Zakazka uzavrena	

## **4 Ověření funkčnosti systému**

Náš informační systém prošel již částečnou testovací fází při vývoji. Během tohoto testování nebyly odhaleny žádné nedostatky a chyby, avšak aby bylo testování kompletní, tak by bylo nutné nasadit ho na skutečná server server a podrobit ho testům nezávislých testerů.

V rámci našich možností bylo testování provedeno a nebyly nalezeny žádné chyby. Předem však nemohu garantovat, že by další testování neodhalilo některé nedostatky.

## **Závěr**

Cílem této bakalářské práce bylo seznámit se s problematikou překladatelských agentur a po jejím prostudování navrhnout informační systém optimální pro nasazení v praxi. Vzhledem k tomu, že nasazení informačních systémů, které by umožňovaly námi vytyčené cíle, v tomto oboru není příliš časté, bylo nutné provést vlastní analýzu problému a zjistit jeho možná východiska. Z této analýzy vznikl systém, který dle mého názoru, poskytuje dobrý základ pro nasazení do praxe. Splňuje všechny požadavky kladené na něj v zadání a umožňuje i několik nadstandardních.

Během vývoje jsem se seznámil s oborem překladatelství, prohloubil své znalosti analýzy a v neposlední řadě rozšířil své povědomí o technologiích .NET. Získal jsem také velmi dobrou představu o náročnosti vývoje informačního systému.

Jeho zlepšení je jistě možné. Sám bych navrhl jeho rozšíření z hlediska zákazníků. Náš informační systém pracuje se zákazníky jako s fyzickými osobami, avšak mezi další možné zákazníky mohou patřit i právnické osoby. Dále by se systém jistě mohl rozšířit o elektronické odesílání dokumentů uvnitř systému, avšak toto by již bylo příliš rozsáhlé pro jednu bakalářskou práci. Proto bych doporučoval prozkoumání těchto možností a případné zamyšlení, jaký by byl přínos těchto změn.

## **Přílohy**

- 1) Uživatelská dokumentace – v elektronické podobě na DVD
- 2) Programátorská dokumentace – v elektronické podobě na DVD
- 3) Vlastní informační systém – na DVD